

# 本周周报(9.9-9.16):

解聪

## 本周工作:

### 1. 淘宝交易地图

本周新添加了两个视窗。并在新加的两个视窗的基础上形成了一个整体的系统。

#### 1.城市的交易矩阵:

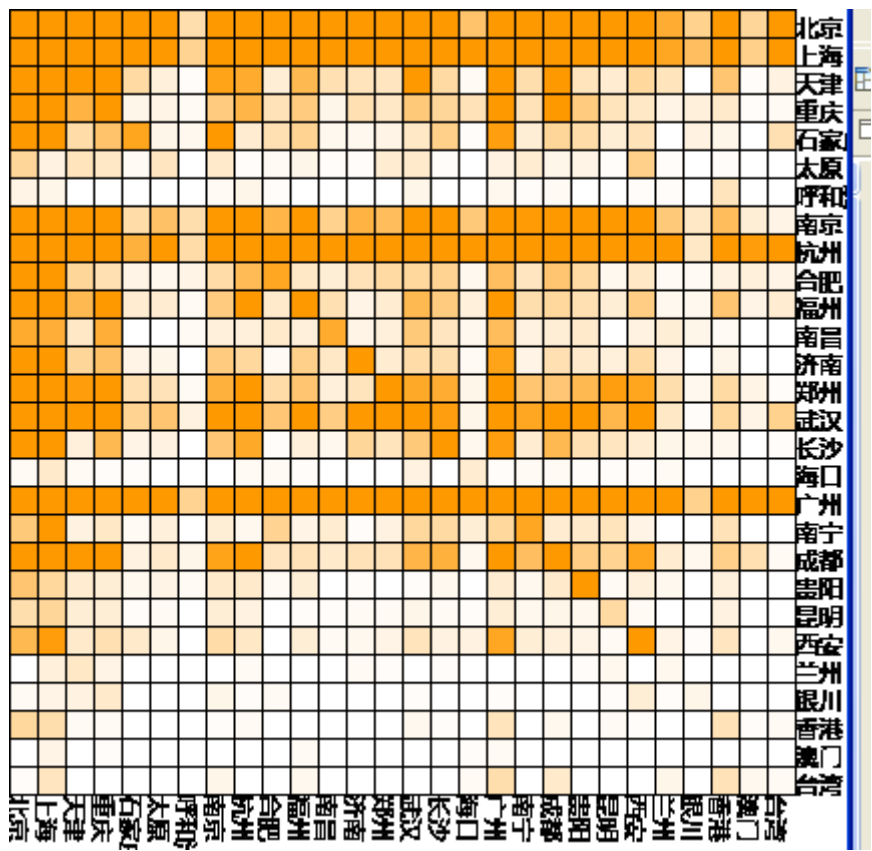


图1 城市交易的矩阵。横向-纵向表示收货-发货的关系。饱和度越低颜色越白，表示交易值越小。饱和度越高颜色越深，表示交易值越大。

#### 2.城市的可视化（意在模仿文字云）:

为了减少交易地图中过于混乱的元素，将各个城市可视化的方法提取出来，组成一个单独的视图。如下图所示:

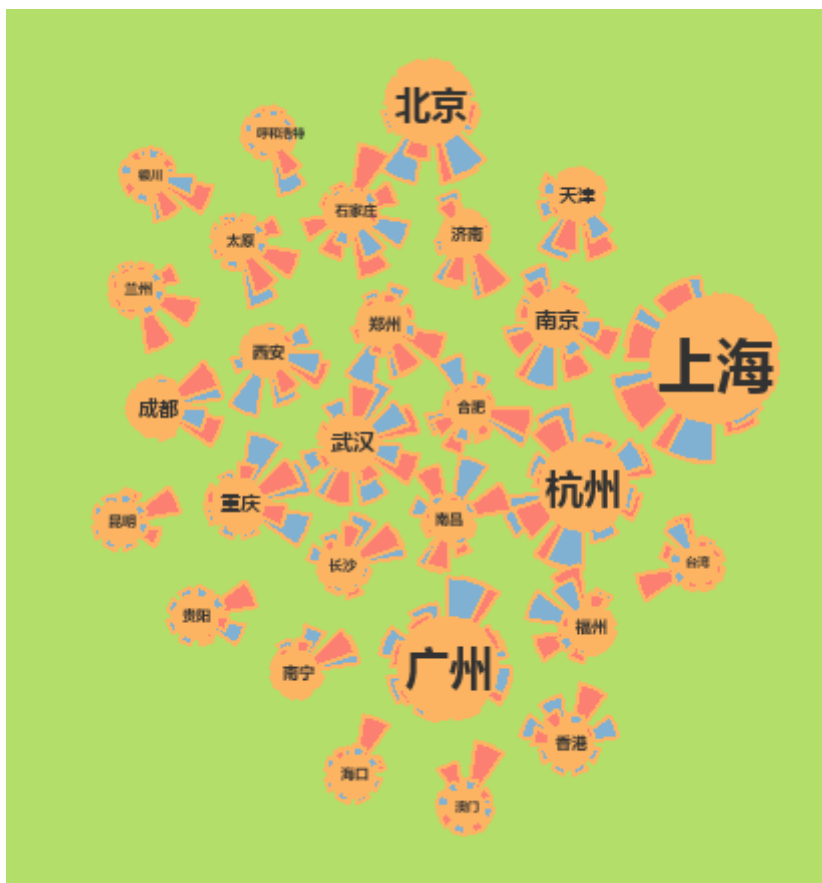


图2.采用各个城市的可视化组成的可视化效果，效果达到实时更新。图中基本保持了各城市相对地理位置不变，又保证各个城市之间互不遮挡。在这样的前提下，城市的花瓣方向仍然用来表示交易的方向。各个城市之间采用紧凑布局，力求达到文本云的一目了然、突出重点的效果。

此可视化主要元素是之前的一级城市可视化。花瓣的方向寓意交易的方向，花瓣的颜色表示进出。对于花瓣的方向，这张图仍然适用，因为其保证了大部分城市之间的相对位置。中间的花心表示绝对交易量的大小。

本方法意图模仿文字云的效果。原本分布密集的城市之间会散开，比如杭州、上海、南京被分散开到一个较大的区域。各个城市基本按照地理位置布局，更新过程中城市相对位置略微有变形。

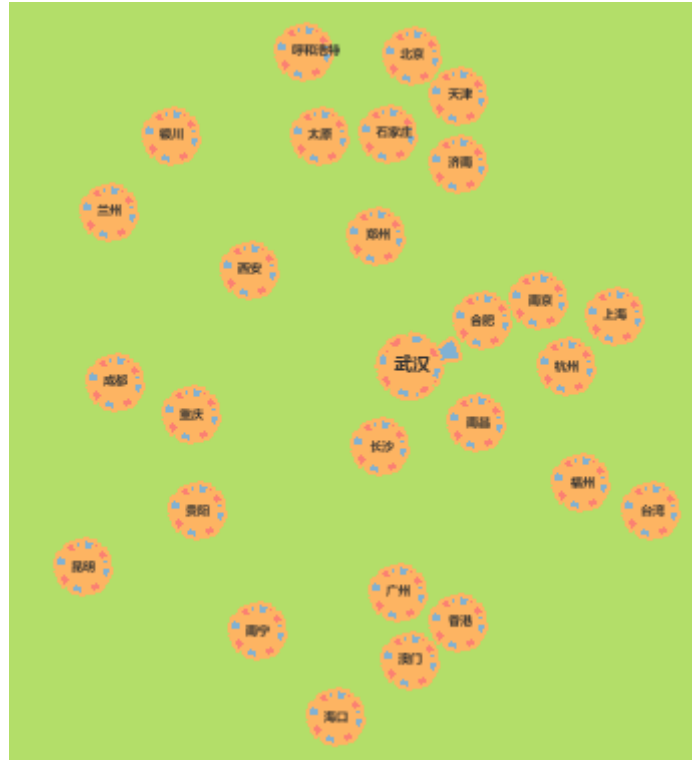


图3.城市的初始状态。各城市按照其地理位置进行初始分布。各城市的半径以及外侧扇形大小均是初始值，随着交易的进行，逐渐调整其位置和大小以达到图2效果。

本算法的实现大致分为以下步骤：

1. 计算各个城市的权重（交易量），并作为其半径大小的依据。
2. 按照权重的大小，权重大的城市优先放置（与地理位置相对应），交易量小的城市放在大城市的空档中间。
3. 如果与先前放置的城市有重叠，则调整其位置直到没有重叠为止。如果不能满足，则调整权重较大城市的位置。略有重叠不影响整个布局。
4. 因为交易在实时更新，重复步骤1，2，3。动态调整城市位置。

算法的实现还是很粗略，布局不够紧凑，而且位置的相对关系有时候不是很准确。本算法从效率与实现效果上需要很大的改进。

同时，图4所展现的信息很大一部分可以从交易密度图中直接获取。比如城市交易量绝对大小，在图4中使用橙色花心大小显示，而在交易地图中使用密度图的高度表示。这样的信息如果使用两个视窗来显示略显多余。

3.交易密度图的修改：

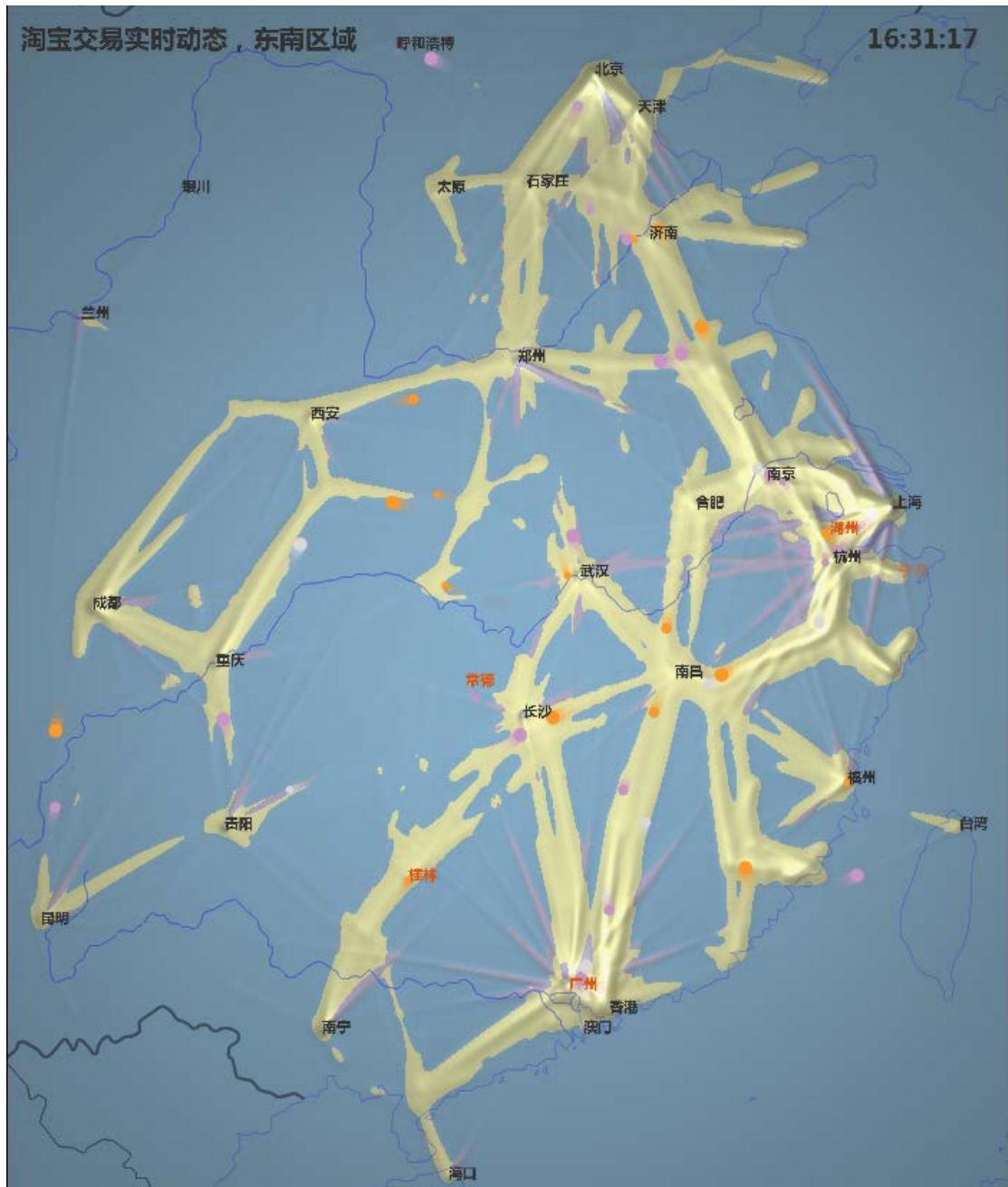


图4.交易地图，调整了对比度，添加了一些动态效果。

交易地图去掉了“花朵”，使其不再遮挡住某个城市的密度图信息。调高了整张图的对比度。添加了一些效果：异地交易时，所交易的城市名闪烁显示（以往只有同城交易的闪烁显示）。

最后采用了一个统一框架将三者整合在一起，初步形成了一个可视化的系统。效果如下：

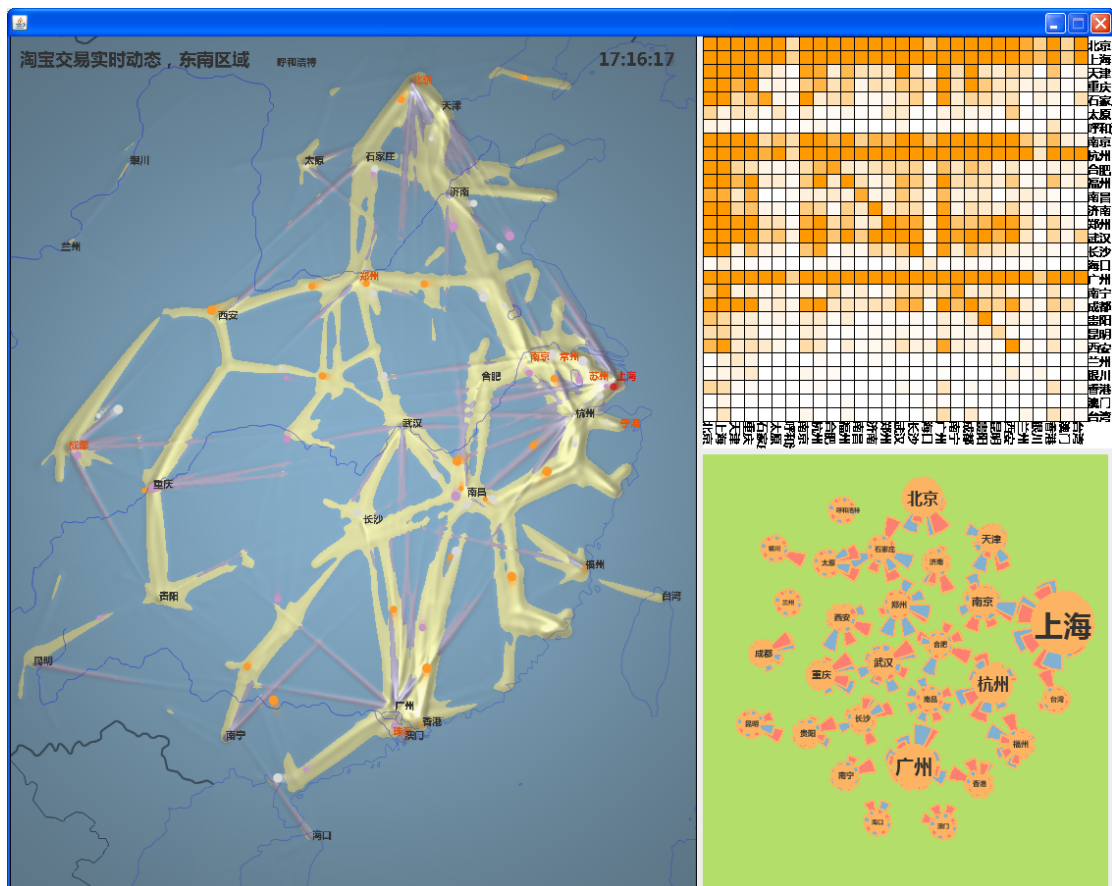


图5.整体效果，包括交易图（左），一级城市交易矩阵（右上）与一级城市的可视化（右下）

另外这周总结了一下交易地图的研究方向。

我觉得城市（接近500个）具有各种属性，某种程度上与单个用户的标签是相对应。比如：某城市都是交易贵重物品，是城市中的“高帅富”。又或者某城市买苹果产品很多，是城市中的“果粉”。

对用户的分析方法可以用来对城市分析。对区域的特征分析也应该建立在对各个城市交易行为的基础上。比如，对应于某一群用户的形象设计，城市也可以有其形象的设计。完全可以将用户的分析手段运用到城市中去。

## 2. DataV 组件

完成了饼状图的基本交互部分。与饼图的落地场景的项目（淘宝数据魔方）了解数据结构与具体需求。

## 下周工作：

### 1. 淘宝交易地图

要明确下一步的方向，还需要和其他人讨论一下，比如从淘宝方面了解除了展示之外其他的准确的需求。与小组其他成员讨论实现一个好的交易地图分析系统所需具备的要素。

完善城市可视化云效果的算法。

添加了新视窗后，感觉程序效率降低了不少，急需整体代码优化与效率的提升。

## **2. DataV 组件的开发**

继续按照组件团队的要求开发和完善现有组件。